Estante Online

# 1. Descrição do Projeto

A Estante *Online* é um programa para ajudar o livreiro e os seus clientes. Através do Programa, um livreiro vai poder cadastrar os livros disponíveis para venda. Há quatro tipos de livros: os de literatura, os técnicos, os periódicos e os de desenvolvimento pessoal. Só há um livreiro, que é como uma espécie de administrador do programa e é responsável por cadastrar as informações dos livros e a quantidade disponível em estoque. Quando o programa inicia pela primeira vez, ele solicita as informações de cadastro do livreiro. Todos os usuários que se cadastrarem depois atuam como clientes desse livreiro. Após se cadastrar no sistema, os clientes podem fazer compras dos livros disponíveis. Além disso, os clientes da livraria possuem uma coleção salva no programa, que é o conjunto de livros que eles possuem. Quando compra um livro pelo sistema, o livro é automaticamente adicionado à coleção do cliente. Entretanto, enquanto visualiza o cadastro do livro, o cliente também pode informar que já o possui e adicioná-lo (manualmente) a sua coleção. O sistema vai gerar vários relatórios importantes, os quais vão ajudar o livreiro a gerenciar seu estoque, e (opcionalmente) também contará com um sistema de recomendação, o qual ajudará os clientes a encontrarem livros interessantes.

# 2. Requisitos Funcionais

1. Cadastrar Livreiro – quando o sistema é iniciado pela primeira vez, ele vai automaticamente perceber que não há um livreiro cadastrado ainda e vai solicitar as informações deste usuário. O livreiro atua como administrador do sistema. Para cadastrar-se como livreiro, é preciso fornecer um endereço de e-mail, uma senha e um nome.

2. Realizar Login – para ter acesso às funcionalidades do sistema, o usuário precisa se autenticar, fornecendo as credenciais de acesso, o e-mail e a senha.

3. Cadastrar Livro – após estar autenticado, o livreiro pode fazer o cadastro de livros que estão à disposição para venda. Há quatro tipos de livros: os de literatura, os técnicos, os periódicos e os de desenvolvimento pessoal. Todo livro possui um título, um resumo, um idioma, um gênero, uma editora e um ano de publicação. O gênero é como se fosse uma categorização do tipo de livro. Os livros de literatura possuem um gênero, o qual pode ser: literatura brasileira, literatura estrangeira, infanto juvenil, artes, biografias ou poesia. O gênero dos períodicos pode ser gibi ou revista de notícias. Os livros de desenvolvimento pessoal podem ser: autoajuda, religião ou saúde. O gênero dos livros técnicos pode ser paradidático ou formação profissional. Os livros de literatura e os de desenvolvimento pessoal possuem um ou mais autores. Os periódicos possuem um mês de lançamento e o número da edição. Os livros técnicos possuem um atributo adicional chamado “assunto”. No cadastro do livro, o livreiro informa também a quantidade de unidades disponíveis e o preço de venda do livro. O livro pode ter outros atributos, além desses (que serão necessários para implementar os demais requisitos – número de visualizações, clientes interessados, lista de comentários, etc.).

4. Editar o Livro – quando o livreiro visualiza as informações detalhadas de um livro, ele pode editar todas as informações do livro, exceto o tipo e o gênero (inclusive, ele pode editar a quantidade disponível em estoque).

5. Cadastrar Cliente – os usuários que se cadastrarem para usar o sistema, após o livreiro, possuirão o perfil de clientes. Os clientes não podem cadastrar livros, como o livreiro. Para cadastrar um cliente, o sistema irá requerer um nome completo, um endereço de e-mail, uma senha, a data de nascimento, o sexo e os três gêneros de livros favoritos.

6. Cadastrar uma nova senha – caso um usuário esqueça a senha de acesso ao sistema, ele pode solicitar recuperar a senha por e-mail. Nesse caso, ele recebe uma senha temporária e, após a usar para logar, automaticamente é direcionado para uma tela onde ele precisará definir uma nova senha.

7. Listar todos os livros – tanto o livreiro quanto os clientes terão acesso a uma lista onde podem ver todos os livros cadastrados no sistema. Esses livros são apresentados numa tabela, cujas colunas apresentam outras informações do livro. Através dessa lista de livros, o usuário (cliente ou livreiro) pode escolher detalhar um livro e ter acesso a mais informações e mais funcionalidades.

8. Filtrar livros por tipo e por palavra-chave – na lista que apresenta todos os livros cadastrados, o usuário (seja cliente ou livreiro) pode realizar a filtragem para exibir livros específicos. Essa filtragem pode levar em consideração o tipo do livro e/ou a presença de uma palavra no título do livro.

9. Ordenar os livros listados em ordem alfabética ou em ordem de avaliação (nota) – na lista que apresenta todos os livros cadastrados, o usuário (seja cliente ou livreiro) pode reordenar a lista para exibir os livros na ordem alfabética ou na ordem de avaliação dos mesmos.

10. Exibir informações detalhadas sobre um livro – a partir da lista de livros, o usuário pode escolher um livro para detalhar. Ao escolher essa opção, ele é direcionado para uma janela com todas as informações do livro. Caso o livreiro acesse essa janela, ele pode: visualizar todos os comentários já feitos sobre o livro, apagar qualquer comentário deixado no livro, visualizar o número de interessados naquele livro, deixar um comentário no livro e fazer uso da função de editar as informações de um livro. Caso o cliente acesse essa janela, ele pode: visualizar todos os comentários já feitos sobre o livro por outros usuários, fazer um comentário sobre o livro, apagar um comentário publicado por ele sobre o livro, indicar que ele está interessado naquele livro (se o livro estiver esgotado), adicionar aquele livro à sua coleção e comprar aquele livro, o que automaticamente também adiciona o livro à sua coleção.

11. Fazer um comentário no cadastro de um livro – quando o usuário (seja o livreiro ou cliente) visualiza o cadastro detalhado do livro, ele pode publicar um comentário sobre o mesmo.

12. Apagar um comentário no cadastro de um livro – quando o cliente visualiza o cadastro do livro, ele pode apagar os comentários que escreveu sobre o mesmo. Quando o livreiro visualiza o cadastro do livro, ele pode apagar qualquer um dos comentários.

13. Indicar interesse em um livro esgotado – quando o cliente visualiza o cadastro do livro e esse livro não possui estoque disponível, surge uma opção para indicar interesse no livro. Indicar interesse significa que o usuário gostaria de comprar aquele livro quando ele estiver disponível novamente. Quando o livro voltar ao estoque (houve novas unidades disponíveis), os usuários que indicaram interesse recebem imediatamente um e-mail informando da disponibilidade.

14. Comprar livro – quando o cliente visualiza o cadastro do livro, ele pode realizar a compra do livro. O sistema vai gerar uma espécie de “boleto” fictício e automaticamente assumir que o usuário já pagou o boleto e comprou o livro. Após comprar um livro, este é adicionado à coleção do cliente.

15. Adicionar livro avulso à coleção – quando o cliente visualiza o cadastro do livro, ele pode adicionar o livro à sua coleção particular. A coleção é uma forma de acessar rapidamente os livros que o usuário já possui.

16. Visualizar os livros da coleção – o cliente pode visualizar os livros que estão na sua coleção. Esses livros são apresentados numa lista, similar a lista que apresenta todos os livros (requisito 7).

17. Avaliar um livro da coleção – os clientes podem avaliar os livros que fazem parte da sua coleção. Essa avaliação pode ser uma nota numérica (na escala que o programador pensar) ou a atribuição de rótulos qualitativos (bom, mediano, ruim, fresco, podre, etc.). Você terá liberdade para aplicar o sistema de avaliação que preferir.

18. Gerar relatório com a lista dos livros com mais pessoas interessadas – o livreiro pode gerar um relatório (em PDF) que vai conter a lista dos 10 livros esgotados que possuem mais pessoas interessadas. Esse é um relatório estratégico que vai ajuda-lo a planejar as próximas compras de livros para a livraria.

19. Gerar relatório com a lista dos livros e suas visualizações – cada vez que um livro for visualizado individualmente (com detalhes – requisito 10), ele incrementa o seu contador de visualizações. O livreiro pode gerar um relatório (em PDF) contendo a lista de todos os livros cadastrados no sistema e o total de visualizações de cada um, ordenado pelo número de visualizações.

20. (desafio I - opcional) Recomendar livros na janela inicial do programa – após realizar login (requisito 2) e ser direcionado para a janela inicial do programa, o usuário visualizaria uma lista de livros personalizada, que seria recomendada pelo programa, com base em algum algoritmo de recomendação (você vai precisar pesquisar sobre como fazer esse sistema de recomendação).

21. (desafio II - opcional) Carregar informações de livros usando um arquivo do Excel – pode ser bastante trabalhoso para o livreiro ter de cadastrar um livro por vez. Assim, essa funcionalidade se refere ao uso de uma planilha com os dados de vários livros (um livro por linha, cada coluna uma informação), que o livreiro usará para cadastrar vários livros de uma vez mais rapidamente.

# 3. Algumas das Classes do Sistema (sugestões)

Usuário – provavelmente, você precisará criar uma superclasse Usuário para concentrar tudo que cliente e livreiro tem em comum. Alguns atributos do usuário serão, por exemplo, um nome, o e-mail e a senha. Os usuários precisarão ter um e-mail único para identificá-los no sistema. Além desses atributos, você pode incluir outros que julgar pertinentes.

Cliente – essa subclasse de usuário concentrará informações pertinentes aos clientes da livraria. O cliente possuirá vários atributos, como uma coleção (uma classe que “conectará” o usuário a uma lista livros – cada item de coleção é composto por um livro, uma avaliação e outros atributos que julgar pertinentes).

Coleção – a coleção representa os livros que o usuário possui e que, portanto, ele pode fazer avaliação. A coleção é composta por itens de coleção. Um item de coleção conecta o usuário, o livro e informações que o usuário queira deixar sobre aquele livro (avaliação e outras coisas que julgar pertinentes).

Avaliação – representa a avaliação deixada pelo cliente para aquele livro (no sistema de avaliação idealizado por você).

Livreiro – essa subclasse de usuário representará o administrador do sistema.

Livro (e suas subclasses) – a superclasse livro reunirá o que todos os livros possuem em comum. Entretanto, vai ser necessário criar uma subclasse para cada tipo de livro existente, para conter os atributos específicos. Além disso, os livros precisarão ter uma lista de visualizações, uma lista de usuários que demonstraram interesse, uma lista de comentários, etc.

Comentário – representa uma mensagem deixada em um livro. Provavelmente, deve ter como atributos a identificação de quem deixou o comentário, a data e a mensagem em si (além de outros atributos que julgar interessantes).

Visualização – quando um cliente visualiza um livro, o sistema guarda essa informação na forma do objeto visualização.

Compra – quando o usuário compra um livro, o sistema guarda essa informação na forma de uma compra.

Observação: Provavelmente, você precisará de uma forma de identificar alguns desses objetos para recuperá-los posteriormente. Quando isso for necessário, sugiro incluir um atributo **id** do tipo **long** no objeto e inicializa-lo via construtor com o **System.currentTimeOnMillis()**. É uma forma simples de criar um identificador único. Outra coisa importante seria pesquisar sobre coleções que existem em Java, além do ArrayList. Há coleções que facilitam bastante a recuperação de informações. Pesquise sobre HashMap e sobre HashSet, por exemplo.

# 4. Como começar esse projeto

* O projeto é algo simples. O que aumenta a sua complexidade são a quantidade de classes envolvidas e como essas classes estarão relacionadas. Em minha opinião, a melhor abordagem para dar início ao projeto é pensar em um modelo de desenvolvimento incremental. Isso significa que você vai adicionando ao projeto uma funcionalidade por vez e, a cada nova adição, você vai mudando no código já escrito as coisas que são necessárias. Dessa forma, o modelo que descreve os seus dados vai mudar bastante ao longo do desenvolvimento, entretanto você vai garantir que estará entregando funcionalidades completas. Alguns alunos acham que é mais interessante pensar em todas as classes de uma vez e em como elas estarão relacionadas. Essa é outra abordagem possível, mas acredito que vai ser bem difícil “acertar” o modelo de dados de primeira, o que pode atrasar bastante o desenvolvimento. Fazendo como eu sugeri acima, você vai conseguir pensar nesse modelo por partes e vai garantir que estará entregando algo ao final da disciplina.
* O primeiro passo, para dar início ao projeto, é resolver a lista de aquecimento. Dessa forma, você já contará com um esqueleto e com a implementação da persistência em XML.
* Acredito que a primeira funcionalidade a ser desenvolvida é o cadastro do administrador (livreiro) e o cadastro dos clientes. Após disponibilizar o cadastro, inclua no projeto a funcionalidade de login. Lembre-se que, dependendo do perfil do usuário, ele deve ser direcionado para uma janela inicial “diferente”. Também é possível já desenvolver a funcionalidade de recuperação de senha.
* Após isso, você pode partir para o desenvolvimento do cadastro dos livros pelo livreiro. Sugiro que deixe disponível o cadastro de todos os tipos de livros disponíveis. Após o cadastro, crie a funcionalidade de listar todos os livros. E, em seguida, a funcionalidade de exibir os detalhes do livro. Lembre-se que o livreiro e o cliente tem acesso a recursos diferentes quando visualizam a janela de detalhamento.
* Após isso, você pode ir para as funcionalidades relacionadas aos comentários nos livros, ao controle das visualizações e marcação de interesse em livros esgotados (visão do cliente). Depois, a funcionalidade do livreiro relacionada ao gerenciamento dos comentários e edição das informações do livro.
* Com tudo isso desenvolvido, você pode começar a parte relacionada às compras e a manutenção das coleções dos clientes.
* Em seguida, você pode aprimorar a listagem dos livros, incluindo a funcionalidade dos filtros e do reordenamento (faça uma pesquisa sobre Comparable e Comparator).
* Por último, você pode fazer as funcionalidades relacionadas à emissão de relatórios.
* Opcionalmente, ao final, você pode começar a se preocupar com os desafios. Pessoalmente, acredito que o desafio II é mais simples que o desafio I. Para fazer o desafio II, pesquise sobre APIs para fazer manipulação de planilhas em Java. Para fazer o desafio I, pesquise sobre Sistemas de Recomendação, os métodos de filtragem (colaborativa, baseada em conteúdo e híbrida).

# 5. Interface Gráfica

O aluno deverá desenvolver o projeto para plataforma desktop (utilizando Swing). **O aluno deve desenvolver a interface gráfica do projeto sem a ajuda de programas do tipo clique e arraste.**

# 6. Persistência de Dados

A persistência de dados pode ser implementada de uma dessas formas: utilizando arquivo (binário, xml, json ou txt) ou banco de dados. No caso de persistência em banco de dados, recomenda-se o uso do JPA. No caso de persistência em formato XML, recomenda-se o uso do XStream.

# 7. Sobre os grupos

Os alunos podem optar em fazer o trabalho individualmente, em dupla ou em trio. A avaliação de cada aluno levará em conta sua contribuição para o projeto final. Mas, caso seja identificado que um aluno não contribuiu o suficiente para o projeto, a nota de toda a equipe poderá ser prejudicada.

# 8. Relatório

## 8.1. Relatório quinzenal

O aluno deverá entregar A CADA 15 DIAS um relatório com ao menos 10 linhas (fonte times new roman 12) explicando:

* O que cada aluno tem feito no projeto;
* Dificuldades;
* Planejamento.

## 8.2. Relatório final

No dia anterior a apresentação, todas as equipes devem DISPONIBILIZAR O EXECUTÁVEL DO PROJETO PARA MIM, para que eu possa me familiarizar com a aplicação, antes do dia da apresentação.

Na apresentação do projeto, CADA ALUNO deve entregar um relatório final, com pelo menos quatro páginas, descrevendo:

* O que o aluno fez no projeto;
* Quais funcionalidades o grupo conseguiu implementar;
* Quais funcionalidades o grupo não conseguiu implementar e por que;
* Experiência do aluno no desenvolvimento no projeto (o que ele mais gostou, maiores dificuldades, se o projeto foi proveitoso para o seu aprendizado).
* Quais as bibliotecas o grupo utilizou para desenvolver o projeto e porque eles optaram por elas;
* Referências que os alunos consultaram para desenvolver o projeto (links de tutoriais, perguntas em fóruns, etc.);
* Impressão de uma classe que os alunos considerem importante no projeto;
* Documentação dessa classe (JavaDoc).

# 7. Sobre a nota

Para a nota do aluno será levado em consideração a quantidade de requisitos funcionais que ele conseguiu implementar e suas decisões em relação aos requisitos não funcionais. Serão levados em consideração também aspectos relacionados ao código e a postura do aluno, em sala e durante a apresentação.

## 7.1. Dimensões que serão avaliadas

|  |
| --- |
| D1 – CÓDIGO |
| D2 – ABSTRAÇÕES OO |
| D3 – COMPLETUDE E CORRETUDE FUNCIONAL |
| D4 – INTERFACE COM USUÁRIO |
| D5 – TRATAMENTO DE SITUAÇÕES DE ERRO |
| D6 – DIFERENCIAL DO PROJETO |
| D7 – AVALIAÇÃO INDIVIDUAL DO ALUNO |
| D8 – TRABALHO EM GRUPO |
| D9 – DEFESA ORAL |

# 8. Sugestões de Referências

A seguir, algumas sugestões de bibliotecas que vocês podem utilizar no projeto.

## 8.1. Geração de Arquivos em PDF

A própria API de manipulação de arquivos nativa de Java pode ser usada para criar arquivos em PDF mais simples. Entretanto, há uma API chamada iText que oferece a mesma função, também é simples, porém mais completa. Usando o IText, vocês podem gerar relatórios que incluam tabelas, cabeçalhos, etc.

Um Bom Tutorial do IText:  
<http://www.forumweb.com.br/artigo/223/java/usando-a-api-itext-para-gerar-documentos-pdf-em-java>

<https://www.youtube.com/watch?v=jcJehf9mGmc>

<https://www.youtube.com/watch?v=eBRL70rR30I>

## 8.2. Persistência de Dados com XStream

Um Bom Tutorial do XStream:  
<http://deixaeuprogramar.weebly.com/home/criando-e-lendo-arquivo-xml-em-java-com-xstream>

<https://youtu.be/jT3FFJFv4dg>

## 8.3. Envio de e-mail

Existem várias APIs para envio de e-mail. Recomenda-se o uso da Java Mail.

Um bom tutorial:

<https://www.devmedia.com.br/enviando-email-com-javamail-utilizando-gmail/18034>

<https://youtu.be/PnIx26AGZ7Y>